

NETWORK PRINT SYSTEM

Patent Number: JP10143339
Publication date: 1998-05-29
Inventor(s): SATO MASAYUKI; SUDO JUNGO; HIRAI NORIO
Applicant(s):: MITSUBISHI ELECTRIC CORP
Requested Patent: ☐ JP10143339
Application Number: JP19960301820 19961113
Priority Number(s):
IPC Classification: G06F3/12 ; B41J5/30 ; B41J29/38
EC Classification:
Equivalents:

Abstract

PROBLEM TO BE SOLVED: To provide a network print system which shortens the time needed for printing even when an abnormality occurs in communications between a client and a server.
SOLUTION: A spooler 12 has a spool file preparation part 15 which divides a print data stream received from a print program 1, prepares a spool file 11 and stores it, a server information management part 18 which manages a server information table 17 which holds transferable and untransferable states, with a server 20, an output destination switching part 19 which switches output destinations when the transfer to the server 20 is not available and a spool file eliminating part 16 which eliminates an unnecessary spool file 11. Every time a spool file 11 is created by the part 15 or each time a spool file is sent from a spool file reinputting part 13 after a fault is recovered, the file 11 is transferred to a spooler 23 of the server 20 and outputted from a printer 21.

Data supplied from the esp@cenet database- 12

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-143339

(43) 公開日 平成10年(1998) 5月29日

(51) Int.Cl. ⁸	識別記号	FI		
G 0 6 F	3/12	G 0 6 F	3/12	D
				B
B 4 1 J	5/30	B 4 1 J	5/30	Z
	29/38		29/38	Z

審査請求 有 請求項の数 9 OL (全 13 頁)

(21) 出願番号 特願平8-301820
(22) 出願日 平成 8 年(1996) 11月13日

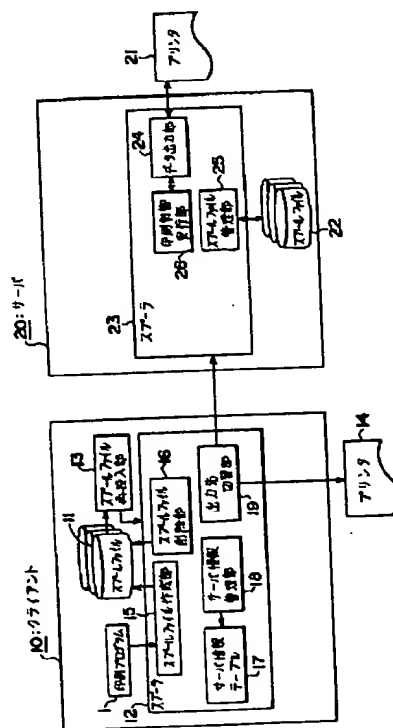
(71) 出願人 000006013
三菱電機株式会社
東京都千代田区丸の内二丁目 2 番 3 号
(72) 発明者 佐藤 雅之
東京都千代田区丸の内二丁目 2 番 3 号 三
菱電機株式会社内
(72) 発明者 須藤 純吾
東京都千代田区丸の内二丁目 2 番 3 号 三
菱電機株式会社内
(72) 発明者 平井 規郎
東京都千代田区丸の内二丁目 2 番 3 号 三
菱電機株式会社内
(74) 代理人 弁理士 吉田 研二 (外 2 名)

(54) 【発明の名称】 ネットワークプリントシステム

(57) 【要約】

【課題】 クライアント・サーバ間の通信で異常が発生した場合でも印刷に要する時間を短縮することのできるネットワークプリントシステムを提供する。

【解決手段】 スプーラ 12 は、印刷プログラム 1 から受け取った印刷データストリームを分割してスプールファイル 11 として作成し保存するスプールファイル作成部 15 と、サーバ 20 との転送可不可状態を保持するサーバ情報テーブル 17 を管理するサーバ情報管理部 18 と、サーバ 20 への転送不可の時出力先を切り替える出力先切替部 19 と、不要となったスプールファイル 11 を削除するスプールファイル削除部 16 と、を有し、スプールファイル 11 がスプールファイル作成部 15 により作成される度に若しくは障害復旧後にスプールファイル再投入部 13 から送られてくる度にスプールファイル 11 をサーバ 20 のスプーラ 23 に転送し、プリンタ 21 から出力させる。



【特許請求の範囲】

【請求項1】 印刷プログラムの実行により得られる印刷データをクライアントのスプーラからサーバのスプーラへ転送し、前記サーバに接続されたプリンタから出力するネットワークプリントシステムにおいて、前記クライアントは、前記印刷プログラムの実行により所定量の印刷データが得られる度にその所定量の印刷データを蓄積するスプールファイルを逐次作成するスプール作成手段を有し、

前記クライアントのスプーラは、前記スプール作成手段により前記スプールファイルが作成される度にそのスプールファイルを前記サーバに転送することを特徴とするネットワークプリントシステム。

【請求項2】 前記クライアントは、前記サーバへの転送が正常に終了した前記スプールファイルを削除する印刷データ削除手段を有することを特徴とする請求項1記載のネットワークプリントシステム。

【請求項3】 前記サーバのスプーラは、前記クライアントから転送されてくる前記スプールファイルを受け取る度にそのスプールファイルに含まれている印刷データを前記プリンタに出力することを特徴とする請求項1記載のネットワークプリントシステム。

【請求項4】 前記サーバは、前記クライアントから送られてくる一連の前記スプールファイルで構成される印刷データが最後まで正常に受信できなかったときに印刷処理を中止させる印刷中止手段を有することを特徴とする請求項1記載のネットワークプリントシステム。

【請求項5】 前記クライアントは、前記スプールファイルを前記クライアントのスプーラに投入する印刷データ再投入手段を有し、

前記クライアントのスプーラは、前記印刷データ再投入手段により前記スプールファイルが投入される度にそのスプールファイルを前記サーバに転送することを特徴とする請求項1記載のネットワークプリントシステム。

【請求項6】 前記クライアントは、前記サーバとの転送可不可状態を保持するサーバ情報保持手段と、クライアント・サーバ間の転送可不可状態を前記サーバ情報保持手段に設定するサーバ情報管理手段と、を有することを特徴とする請求項1記載のネットワークプリントシステム。

【請求項7】 前記サーバ情報管理手段は、クライアント・サーバ間通信に異常が発生したときに当該サーバとの転送が不可である旨を前記サーバ情報保持手段に設定することを特徴とする請求項6記載のネットワークプリントシステム。

【請求項8】 前記サーバ情報管理手段は、クライアント・サーバ間の通信障害が回復したときに当該サーバとの転送が可能である旨を前記サーバ情報保持手段に設定することを特徴とする請求項6記載のネットワークプリントシステム。

ントシステム。

【請求項9】 前記クライアントは、前記サーバへの転送が不可能なときに前記スプールファイルの転送先を前記サーバのスプーラから他へ切り替える転送先切替え手段を有することを特徴とする請求項1記載のネットワークプリントシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、クライアント及びサーバにより構築されたネットワークシステムにおいて、印刷データをクライアントのスプーラからサーバのスプーラへ転送し、サーバに接続されたプリンタから出力させるネットワークプリントシステムに関する。

【0002】

【従来の技術】図16は、例えば、日経BP社刊Net Wareシステム構築技法に記載された従来のネットワークプリントシステムの全体構成図である。クライアント100は、印刷プログラム101の実行により送られてくる印刷データストリームを受け取りサーバ200へ転送するスプーラ102を有している。スプーラ102は、受け取った印刷データストリームを転送可能な形式にデータ変換してスプールファイルとしてを作成するスプール作成部103と、接続したサーバ200に対して印刷データ（スプール）を送出するデータ出力部104とを有している。なお、転送されるスプールは、厳密にはファイルとして作成されるが、そのファイルは、スプーラ102の内部において便宜上作成されるだけであり外部に保存されることはないため、ファイルではなく単にスプールという表現を用いることにする。

【0003】一方、サーバ200は、プリンタ201と、印刷データを一時保存するスプールファイル202と、クライアント100から送られてくる印刷データを受け取りプリンタ201に出力するスプーラ203とを有している。スプーラ203は、印刷データをプリンタ201に出力するデータ出力部204と、印刷データを印刷可能な形式で保存するスプールファイル202を作成するとともに不要となったスプールファイル202を削除するスプールファイル管理部205を有している。

【0004】以上の構成を有する従来例の印刷処理を以下に説明するが、最初に印刷データの転送が正常に終了した場合における処理から説明する。図17に印刷処理が正常に終了したときのシステムタイムチャートを示す。

【0005】印刷プログラム101が実行されることにより作成された印刷データストリームを受け取ると、スプール作成部103は、印刷データストリームからスプーラを作成する。スプーラ102は、作成したスプーラをデータ圧縮など所定の転送形式に変換した後にデータ出力部104から出力ポートを介してサーバ200へ転送する。

【0006】サーバ200において、スプールファイル管理部205は、受け取ったスプールを保存するためにスプールファイル202を作成し、印刷可能な形式で保存する。スプーラ203は、スプールファイル202に保存した印刷データの出力処理をスケジューリングし、プリンタ201が未使用であることを確認した後スプールの印刷を開始する。印刷が終了すると、スプールファイル管理部205は、印刷が終了した印刷データを保存するスプールファイル202を削除する。

【0007】次に、クライアント・サーバ間通信に異常が検出されたときの印刷処理について説明する。この異常が検出されたときのシステムタイムチャートを図18に示す。

【0008】正常時の場合と同様に印刷データストリームが作成され、データ出力部104から最初のスプールをサーバ200へ転送しようとしたときに障害が検出されると、その障害が復旧されるまで待ち、障害からの回復後にスプールをサーバ200へ転送する。サーバ200においては、正常時と同様に動作し、受け取ったスプールをスプールファイル202に保存し、その保存した印刷データを出力する。そして、印刷が終了した印刷データを保存するスプールファイル202を削除する。

【0009】また、特開平8-6747号公報には、クライアント側で印刷データと印刷制御情報を蓄積保存する手段を設けた構成が開示されており、この従来例においては、印刷プログラムが作成する印刷データをその蓄積保存する手段に保存した時点で印刷プログラムの処理終了とすることで印刷プログラムの処理速度を短縮でき、また、発生した障害の回復後に保存した印刷データの再転送ができるようにしている。

【0010】

【発明が解決しようとする課題】しかしながら、従来においては、サーバに転送すべき印刷データを全て受け取った後にサーバへの転送処理を行うようにしているため、あまり効率的な印刷処理であるとはいえない。すなわち、例えば、印刷データを作成する処理が1時間要するとした場合、この処理の初期に作成された印刷データは、印刷可能であるにもかかわらず少なくとも1時間待たされることになる。従って、より効率的にかつ印刷処理に要する時間を短縮できるシステムが望まれる。

【0011】また、クライアント・サーバ間通信に異常が生じているとした場合、従来においては、印刷データの作成後に行うサーバへの転送処理のときになって初めて通信障害を検出することになる。すなわち、例えば、印刷データを作成する処理が1時間要するとした場合、障害の検出は、少なくとも1時間経過した後に初めて気付くことになる。従って、障害が発生したときでも印刷処理に要する時間を短縮できるシステムが望まれる。

【0012】本発明は以上のような問題を解決するためになされたものであり、第1の目的は、サーバ側での印

刷開始をより早く行えるようにすることで印刷に要する時間を短縮することのできるネットワークプリントシステムを提供することにある。

【0013】また、第2の目的は、クライアント・サーバ間の通信で異常が発生した場合でもその障害を迅速に検出することによってサーバ側での印刷開始をより早く行えるようにすることで印刷に要する時間を短縮することのできるネットワークプリントシステムを提供することにある。

【0014】

【課題を解決するための手段】以上のような目的を達成するために、第1の発明に係るネットワークプリントシステムは、印刷プログラムの実行により得られる印刷データをクライアントのスプーラからサーバのスプーラへ転送し、前記サーバに接続されたプリンタから出力するネットワークプリントシステムにおいて、前記クライアントは、前記印刷プログラムの実行により所定量の印刷データが得られる度にその所定量の印刷データを蓄積するスプールファイルを逐次作成するスプール作成手段を有し、前記クライアントのスプーラは、前記スプール作成手段により前記スプールファイルが作成される度にそのスプールファイルを前記サーバに転送するものである。

【0015】第2の発明に係るネットワークプリントシステムは、第1の発明において、前記クライアントは、前記サーバへの転送が正常に終了した前記スプールファイルを削除する印刷データ削除手段を有するものである。

【0016】第3の発明に係るネットワークプリントシステムは、第1の発明において、前記サーバのスプーラは、前記クライアントから転送されてくる前記スプールファイルを受け取る度にそのスプールファイルに含まれている印刷データを前記プリンタに出力するものである。

【0017】第4の発明に係るネットワークプリントシステムは、第1の発明において、前記サーバは、前記クライアントから送られてくる一連の前記スプールファイルで構成される印刷データが最後まで正常に受信できなかったときに印刷処理を中止させる印刷中止手段を有するものである。

【0018】第5の発明に係るネットワークプリントシステムは、第1の発明において、前記クライアントは、前記スプールファイルを前記クライアントのスプーラに投入する印刷データ再投入手段を有し、前記クライアントのスプーラは、前記印刷データ再投入手段により前記スプールファイルが投入される度にそのスプールファイルを前記サーバに転送するものである。

【0019】第6の発明に係るネットワークプリントシステムは、第1の発明において、前記クライアントは、前記サーバとの転送不可状態を保持するサーバ情報保持手段と、クライアント・サーバ間の転送不可状態を

前記サーバ情報保持手段に設定するサーバ情報管理手段とを有するものである。

【0020】第7の発明に係るネットワークプリントシステムは、第6の発明において、前記サーバ情報管理手段は、クライアント・サーバ間通信に異常が発生したときに当該サーバとの転送が不可である旨を前記サーバ情報保持手段に設定するものである。

【0021】第8の発明に係るネットワークプリントシステムは、第6の発明において、前記サーバ情報管理手段は、クライアント・サーバ間の通信障害が回復したときに当該サーバとの転送が可能である旨を前記サーバ情報保持手段に設定するものである。

【0022】第9の発明に係るネットワークプリントシステムは、第1の発明において、前記クライアントは、前記サーバへの転送が不可能なときに前記スプールファイルの転送先を前記サーバのスプーラから他へ切り替える転送先切替え手段を有するものである。

【0023】

【発明の実施の形態】以下、図面に基いて、本発明の好適な実施の形態について説明する。

【0024】図1は、本発明に係るネットワークプリントシステムの一実施の形態を示した全体構成図である。図1には、印刷を依頼するクライアント10と印刷を請け負うサーバ20とが接続された構成が示されている。なお、クライアント10は、複数台のサーバ20に接続可能であり、サーバ20は、複数台のクライアント10に接続可能であるが、便宜上1台ずつ図示する。

【0025】クライアント10は、分割された所定量の印刷データを保存するスプールファイル11と、印刷プログラム1の実行により送られてくる印刷データストリームを受け取りサーバ20へ転送するスプーラ12と、スプールファイル11をクライアント10のスプーラ12に投入する印刷データ再投入手段としてのスプールファイル再投入部13と、ローカルに接続されたプリンタ14とを有している。スプーラ12には、スプールファイル作成部15、スプールファイル削除部16、サーバ情報テーブル17、サーバ情報管理部18及び出力先切替部19が含まれている。スプールファイル作成部15は、スプール作成手段として設けられ、印刷プログラム1の実行により所定量の印刷データが得られる度にその所定量の印刷データを蓄積するスプールファイル11を逐次作成する。スプールファイル削除部16は、印刷データ削除手段として設けられ、サーバ20への転送が正常に終了したスプールファイル11を削除する。サーバ情報テーブル17は、サーバ情報保持手段として設けられ、サーバ20との転送不可状態を保持している。図2は、サーバ情報テーブル17の設定内容例を示した図であり、サーバ毎に転送可能フラグを設定する。本実施の形態では、転送可能フラグが“0”のとき転送可能とし、“1”のとき転送不可としている。サーバ情報管理

部18は、サーバ情報管理手段であり、サーバ20への転送不可状態に応じてサーバ情報テーブル17の転送可能フラグの設定を行う。すなわち、クライアント・サーバ間通信に異常が発生したときに転送不可である旨である“1”を、通信障害が回復したときに転送が可能である旨である“0”を、それぞれそのサーバ20に対応した転送可能フラグに設定する。出力先切替部19は、転送先切替え手段として設けられ、サーバ20への転送が不可能なときにスプールの転送先をサーバ20のスプーラから他へ切り替える。本実施の形態における出力先切替部19は、図1に示したようにローカルに接続したプリンタ14に出力するようにしている。

【0026】一方、サーバ20は、プリンタ21と、クライアント10から転送されてくるデータを一時保存するために作成されるスプールファイル22と、クライアント10からスプールファイルを受け取る度にプリンタ21に出力するスプーラ23とを有している。スプーラ23には、データ出力部24、スプールファイル管理部25、印刷制御実行部26とが含まれている。データ出力部24は、印刷データをプリンタ21に出力する。スプールファイル管理部25は、印刷可能な形式でスプールファイル22を作成するとともに不要となったスプールファイル22を削除する。印刷制御実行部26は、印刷中止手段として設けられており、クライアント10から送られてくる一連のスプールファイルで構成される印刷データが最後まで正常に受信できなかったときに印刷処理を中止させる。

【0027】なお、上記構成においてスプールファイル11、22は、各計算機10、20に接続されたディスク装置(図示せず)に作成され、その他の構成である印刷プログラム1、スプーラ12、23、スプールファイル再投入部13は、ソフトウェアで作成され、各計算機10、20のCPU上で各機能を発揮することになる。

【0028】本実施の形態において特徴的なことは、クライアント10側で作成された印刷データを所定量毎に分割してスプールファイル11を作成し、スプールファイル11が作成される度にそのスプールファイル11をサーバ20側に転送するようにしたことである。すなわち、クライアント10において印刷データストリームに基づくスプールファイル11の作成処理と、作成されたスプールファイル11のサーバ20への転送処理とを並行して実行できるようにしたことである。これにより、全体を通して印刷処理に要する時間を大幅に短縮することができるようになる。

【0029】次に、本実施の形態における印刷処理について説明する。最初に、新規に印刷データが作成され、その印刷データが正常にサーバ20に転送されプリンタ21から印刷される場合の処理について説明する。なお、図3は、印刷処理が正常に終了したときのタイムチャートを示した図であり、従来例の図17に対応する。

【0030】まず、図4に示したように、印刷プログラム1は、出力先となるサーバ20及びそのサーバ20に接続されているプリンタ21が指定されると、印刷データの作成処理が開始され（ステップ101）、印刷データストリームを作成する（ステップ102）。そして、作成した印刷データストリームをスプーラ12に順次送出する（ステップ103）。

【0031】図5は、スプーラ12が行う印刷処理を示したフローチャートである。図5によると、スプーラ12は、イベントの発生待ち状態にある（ステップ111）。ここで、印刷プログラム1が実行されたことにより印刷データストリームが送られてくると（ステップ112）、スプールファイル11の作成/転送処理が実行される（ステップ113）。このスプールファイル11の作成/転送処理の詳細について図6に示したフローチャートを用いて説明する。

【0032】まず、この処理を実行するスプールファイル作成/転送ルーチンは、印刷データストリームが発生した場合に呼ばれるルーチンである。まず、変数*i*、Retに1を代入することによって初期化する（ステップ121）。スプールファイル作成部15は、印刷データストリームを逐次受け取ると、所定量の大きさを持つバッファ（図示せず）に読み込む（ステップ122）。そして、バッファサイズ分の読み込んだ印刷データをネットワーク送信が可能な所定の形式に変換し、スプールファイル11として作成して保存する（ステップ123）。このスプールファイル11が1つ作成された時点で図7に示した転送可能チェック処理を行う（ステップ124）。なお、本実施の形態においては、印刷データストリームが分割されるときに所定量は、バッファの大きさに依存することになるが、転送サイズを考慮するなどしてバッファサイズに依存しない任意の所定量を設定したり、マルチバッファを用いたりしてもよい。所定量の設定は、単なる設計事項である。

【0033】転送可能チェックルーチンは、サーバ情報テーブル17の転送可能フラグを参照することによって転送可能かどうかの判断を行う（ステップ131）。本実施の形態では、サーバ情報テーブル17から読み取った値をそのままスプールファイル作成/転送ルーチンに返す（ステップ132、133）。

【0034】図6に示したスプールファイル作成/転送処理において、スプールファイル11の転送が可能な状態（転送可能フラグ＝“0”）であるとき、作成したスプールファイル11をサーバ20へ転送する（ステップ125）。転送が正常に終了したとき、変数*i*をインクリメントする（ステップ126）。

【0035】一方、スプールファイル11の転送が可能な状態でない（転送可能フラグ＝“1”）とき、サーバ情報管理部18に、サーバ情報テーブル17の転送先のサーバ20に対応する転送可能フラグの値に“1”をセ

ットさせる（ステップ127）。そして、本ルーチンのリターンステータスRetに異常を表す“0”を代入し（ステップ128）、変数*i*をインクリメントする（ステップ126）。なお、以上の処理から明らかなように、変数*i*は、印刷データストリームの分割数、すなわち作成されるスプールファイル11の総数を算出するための変数である。

【0036】上記処理後、バッファへの読み込みが印刷データストリームの終わりまで達していないときには印刷データストリームのバッファへの読み込み処理（ステップ122）に戻る（ステップ129）。印刷データストリームの終わりに達したとき、グローバル変数ALLに（*i*－1）を代入してリターンステータスRetを返す（ステップ130）。なお、グローバル変数ALLの値は、印刷データストリームを分割することにより作成されたスプールファイルの総数である。

【0037】以上のようにして、本実施の形態によれば、図5に示したスプールファイル作成/転送処理（ステップ113）において、作成された印刷データストリームに基づいてスプールファイル11が作成される度にサーバ20に逐次転送することになる。すなわち、図3から明らかなように、本実施の形態では、バッファが印刷データストリームの一部で満たされスプールファイル11が作成される時間のみ待つだけでサーバ20への転送処理が開始されるので、印刷処理時間を大幅に短縮することができる。

【0038】図5において、スプールファイル11の転送/作成処理が、異常終了したとき（Ret＝0）には、イベントの待ち状態に戻り（ステップ111）、正常終了したとき（Ret＝1）には、スプールファイル11の削除を行う（ステップ114）。このステップ114におけるスプールファイル削除部16が行うスプールファイル削除処理について図8に示したフローチャートを用いて説明する。

【0039】スプールファイル削除ルーチンでは、変数*i*を0で初期化した後（ステップ141）、スプールファイル（*i*）を削除する（ステップ142）。変数*i*をインクリメントし（ステップ143）、グローバル変数ALLに変数*i*が達するまで上記処理を繰り返し行う（ステップ144）。これにより、クライアント10において作成された全スプールファイル11を削除することができる。

【0040】次に、スプールファイル11が転送されてくるサーバ20における印刷処理について図9を用いて説明する。まず、クライアント10から一連のスプールファイルを受け取れたときの正常時における処理から説明する。

【0041】サーバ20におけるスプーラ23は、スプールファイルを受け取ると、それをサーバ20側のスプールファイル22として保存する（ステップ151）。

そして、スプール23は、最初のスプールファイルを受け取ったときにプリンタ21が印刷要求待ち状態であればスプールファイル22に含まれている印刷データの印刷を開始する(ステップ152)。印刷要求待ち以外の状態であれば、プリンタ21への出力処理をスケジューリングしておく。そして、クライアント10から最終のスプールファイルが転送されてくるまで上記処理を繰り返し行う(ステップ154)。スプール23が最終のスプールファイルを受け取り、プリンタ21が印刷を正常に終了すると、その旨をクライアント10に通知する(ステップ155)。

【0042】ところで、本実施の形態では、1つの印刷データストリームを複数のスプールファイル11に分割してサーバ20に転送するようにしているが、各スプールファイル11には、後続するスプールファイル11の存否を表す情報が含まれている。従って、サーバ20のスプール23は、この情報をスプールファイルの受信の度に確認することによって後続するスプールファイルの存否及び最後尾であるかどうかを判断することができる。これにより、ステップ152において、本来後続して転送されてくるはずのスプールファイルが所定時間内に受信しなかったときは、何らかの通信障害が発生したとして印刷を中止することになる(ステップ156)。ここで、印刷制御実行部26が行う印刷を中止する処理の詳細を図10を用いて説明する。

【0043】印刷制御実行部26は、タイムアウトの発生時、受信できなかったスプールファイルと同じ印刷データストリームを構成する他のスプールファイル22を印刷しているかどうかを調べる(ステップ161)。印刷中のものがあれば、プリンタ21に印刷中止コードを送出し、強制的に印刷を中止させる(ステップ162)。一方、印刷中のものがなければ、既に受け取っている他のスプールファイル22は、まだ印刷されていないと判断してスケジュールから削除する(ステップ163)。その後、1つの印刷データストリームを構成する一連のスプールファイル22を全て削除する(ステップ164)。異常終了のため、正常に受け取った一連のスプールファイル22を保持していても無駄だからである。但し、一連のスプールファイル22に通し番号が付されていて、サーバ20からクライアント10に再送して欲しい通し番号を通知するなどすれば、受信済みのスプールファイル22を削除しない方が効率的である。このようにして、クライアント・サーバ間通信に障害が発生した場合にもサーバ20において対処することができる。

【0044】本実施の形態では、以上のように所定量の印刷データストリームが作成される度にサーバ20にスプールファイルを送るようにした。図3と図17とを比較すると、クライアント10側におけるジョブのみならず、印刷処理に要する時間を大幅に短縮できたことが

目瞭然である。もちろん、印刷データの作成、データ転送、受信、印刷それぞれに要する時間は、従来例と同様である。本実施の形態においては、印刷データを分割してスプールファイルを作成することによってクライアント10におけるスプールファイルの作成処理と転送処理とを並行してできるようにしたので、印刷処理に要する時間を大幅に短縮することができる。

【0045】以上の処理は、基本的に印刷データが正常にサーバ20に転送されプリンタ21から印刷される場合の処理について説明したが、次に、作成したスプールファイル11がサーバ20に転送されなかったときの異常時における処理について説明する。なお、図11は、クライアント・サーバ間の転送処理に異常を検出したときのシステムタイムチャートを示した図であり、従来例の図18に対応する。

【0046】図5において、スプール12は、イベントの発生の待ち状態にあるときに印刷データストリームが送られてくると、スプールの作成/転送処理が実行される(ステップ111~113)。このスプールの作成/転送処理の詳細については、図6に示したフローチャートを用いてすでに説明した。ところで、図6におけるステップ124、つまり図7に示した転送可能チェック処理により指定のサーバ20との通信が可能であるかどうかを判断しているが、この転送可能チェック処理は、最初のスプールファイル11が作成された時点で実行される。すなわち、従来においては、印刷データストリーム全体が作成されるのを待って初めて転送先となるサーバ20との通信が可能かどうかの判断をしていたのに対し、本実施の形態においては、印刷データストリームの一部すなわち最初のスプールファイル11が作成され転送しようとした時点でその判断を行うことができる。

【0047】以上のように、サーバ20への転送ができない状態にあるとき、すなわちステップ113におけるスプール作成/転送処理からのリターンステータスが異常終了を表す“0”であったとき、イベントの発生の待ち状態に戻るが(ステップ111)、転送が実行できない間にも残りのスプールファイル11の作成処理は、継続して行われる。

【0048】ところで、サーバ情報管理部18は、自らサーバ間の通信状態を定期的に監視し、又は間接的にシステム管理者等による設定入力により障害が復旧されたことを検出すると、当該サーバ20に対応したサーバ情報テーブル17の転送可能フラグを“0”にセットする。図12にスプールファイル再投入部13における処理のフローチャートを示すが、スプールファイル再投入部13は、システム管理者等により起動され、又は障害時にサーバ情報テーブル17の転送可能フラグを常時監視し、当該サーバ20の転送可能フラグが“0”にセットされたことを検出すると、通信が復旧したと認識し(ステップ171)、スプール12に作成されているス

プールファイル11を順次送出する(ステップ172)。

【0049】図5において、イベント待ちしていたスプーラ12は、スプールファイル再投入部13からスプールファイル11を受け付けると(ステップ111、112、115)、スプールファイル転送処理を行う(ステップ116)。この処理は、図6に示したスプールファイル作成/転送処理からスプールファイル作成処理を排除したのとほぼ同様になる。このスプールファイル転送処理のフローチャートを図13に示す。

【0050】図13において、スプーラ12は、まず、変数*i*、Retに1を代入することによって初期化する(ステップ181)。そして、作成されているスプールファイル11をサーバ20へ転送する(ステップ182)。転送が正常に終了したときには、変数*i*をインクリメントする(ステップ183)。この処理を変数*i*がグローバル変数ALLに達するまで上記処理を繰り返す(ステップ184)。

【0051】一方、スプールファイル11の転送が失敗したとき、サーバ情報管理部18に、サーバ情報テーブル17の転送先のサーバ20に対応する転送可能フラグの値に“1”をセットさせる(ステップ185)。そして、本ルーチンのリターンステータスRetに異常を表す“0”を代入する(ステップ186)。そして、スプールファイル11の転送が正常に実行でき印刷データストリームの末尾に達したとき、あるいはスプールファイル11の転送が正常に実行できなかった時点でリターンステータスRetを返し(ステップ187)、本処理を終了する。

【0052】以上のように、本実施の形態によれば、印刷データストリームをスプールファイル11に保存するようにしたので、障害から復旧した時点で印刷プログラム1を再起動しなくても作成されたスプールファイル11をサーバ20に転送することができる。そして、図5におけるスプールファイル転送処理(ステップ116)の後、前述したスプールファイル削除処理(ステップ114)を行う。

【0053】図11から明らかなように、本実施の形態では、1つのスプールファイル11の作成完了後にサーバ20への転送処理を開始するようにしているので、その時点でクライアント・サーバ間通信の異常を検出することができる。これにより、印刷データストリームからスプールファイル11全ての作成を待たずにそのときからすぐに障害の復旧作業に取りかかることができる。また、その時間分、サーバ20への転送処理も早く再開することができるので、印刷処理に要する時間を大幅に短縮することができる。

【0054】ところで、サーバ情報管理部18がサーバ情報テーブル17の設定値を変更することは、上述した通りであるが、このサーバ情報管理部18において行わ

れる処理について図14に示したフローチャートを用いて簡単に説明すると、サーバ情報管理部18は、サーバ名と更新すべき転送可能フラグの値を受け付けると(ステップ191)、サーバ情報テーブル17の指定されたサーバ20の転送可能フラグに指定の転送可能フラグ値をセットする(ステップ192)。このようにして、クライアント10が接続されている各サーバへの転送が可能かどうかをサーバ情報テーブル17に設定するようにしたので、その判断を容易に行うことができる。

【0055】また、本実施の形態では、出力先切替部19を設けたことも特徴の一つとしている。この出力先切替部19における処理について図15に示したフローチャートを用いて説明する。

【0056】出力先切替部19は、スプールファイル1の出力先すなわちサーバ名を受け付けると(ステップ201)、サーバ情報テーブル17のそのサーバ20に対応した転送可能フラグの値をサーバ情報テーブル17から受け取る(ステップ202)。その値が転送可能(転送可能フラグ=“0”)である場合、出力先をサーバ20に向ける(ステップ203)。転送不可(転送可能フラグ=“1”)である場合、出力先をプリンタ14が接続されているローカルポートに向ける(ステップ204)。これにより、サーバ20に接続されているプリンタ21からの出力が不可能でときであつてもローカルに接続した代わりのプリンタ14から印刷データを出力させることができる。

【0057】なお、本実施の形態では、サーバ20への転送が不可能なときには無条件にローカルのプリンタ14へ出力を切り替えるようにしたが、ローカルのプリンタ14でなくても予備的な他のサーバへ切り替えるようにしてもよい。また、切替え先を設定したテーブルなどを持つことで、任意のサーバへ出力先を切り替えるようにすることもできる。

【0058】

【発明の効果】本発明によれば、クライアント側で作成された印刷データを所定量毎に分割してスプールファイルを作成し、スプールファイルが作成される度にそのスプールファイルをサーバ側に転送するようにした。すなわち、スプールファイルが作成される時間のみ待てば、サーバへの転送処理が開始され、その後は、印刷データに基づくスプールファイルの作成処理と、作成されたスプールファイルのサーバへの転送処理とを並行して実行できるようにしたので、印刷処理全体に要する時間を大幅に短縮することができる。

【0059】また、印刷データ削除手段を設けたので、転送が正常に終了したスプールファイルを削除することができる。

【0060】また、サーバ側において、転送されてくるスプールファイルを受け取ると即座に印刷を開始するようにしたので、スプールファイルの受信処理とプリンタ

10

20

30

40

50

への出力処理とを並行して行うことができる。これにより、印刷処理全体に要する時間を大幅に短縮することができる。

【0061】また、サーバ側において、転送されてくるはずのスプールファイルを正常に受け取ることができなかったとき、そのスプールファイルと同じ印刷データストリームを構成する他のスプールファイルが印刷途中であればそれを中止し、まだ印刷されていないときはスケジュールから削除するようにしたので、無駄な印刷をしなくてすむ。

【0062】また、印刷データ再投入手段を設けたことにより、障害が発生しその復旧をした後に、作成され保存されたままのスプールファイルの転送を再開することができる。

【0063】また、サーバ情報管理手段を設け、クライアントが接続されている各サーバへの転送が可能かどうかをサーバ情報保持手段に設定するようにしたので、転送不可の判断を容易に行うことができる。

【0064】また、転送先切替手段を設けたので、本来印刷を行うべきサーバとの通信が正常にできない場合であっても他のサーバやローカルのプリンタへその出力先を切り替えることができる。

【図面の簡単な説明】

【図1】 本発明に係るネットワークプリントシステムの一実施の形態を示した全体構成図である。

【図2】 本実施の形態におけるサーバ情報テーブルの設定内容例を示した図である。

【図3】 本実施の形態において印刷処理が正常に終了したときのタイムチャートを示した図である。

【図4】 本実施の形態における印刷プログラムにおける処理を示したフローチャートである。

【図5】 本実施の形態におけるクライアントのスプーラが行う印刷処理を示したフローチャートである。

【図6】 図5に示したスプールファイル作成/転送処理の詳細を示したフローチャートである。

【図7】 図6に示した転送可能チェック処理を示した

フローチャートである。

【図8】 図5に示したスプールファイル削除部が行うスプールファイル削除処理を示したフローチャートである。

【図9】 本実施の形態におけるサーバのスプーラが行う印刷処理を示したフローチャートである。

【図10】 図9に示した印刷制御実行部が行う印刷中止処理を示したフローチャートである。

【図11】 本実施の形態においてクライアント・サーバ間の転送処理に異常が検出されたときのシステムタイムチャートを示した図である。

【図12】 本実施の形態のスプールファイル再投入部における処理を示したフローチャートである。

【図13】 図5に示したスプールファイル転送処理を示したフローチャートである。

【図14】 本実施の形態のサーバ情報管理部において行われる処理を示したフローチャートである。

【図15】 本実施の形態の出力先切替部において行われる処理を示したフローチャートである。

【図16】 従来のネットワークプリントシステムを示した全体構成図である。

【図17】 従来において印刷処理が正常に終了したときのタイムチャートを示した図である。

【図18】 従来においてクライアント・サーバ間の転送処理に異常が検出されたときのシステムタイムチャートを示した図である。

【符号の説明】

1 印刷プログラム、10 クライアント、11、22 スプールファイル、12、23 スプーラ、13 スプールファイル再投入部、14、21 プリンタ、15 スプールファイル作成部、16 スプールファイル削除部、17 サーバ情報テーブル、18 サーバ情報管理部、19 出力先切替部、20 サーバ、24 データ出力部、25 スプールファイル管理部、26 印刷制御実行部。

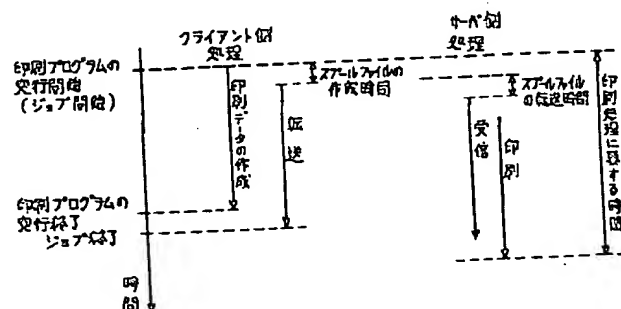
【図2】

サーバ情報テーブル

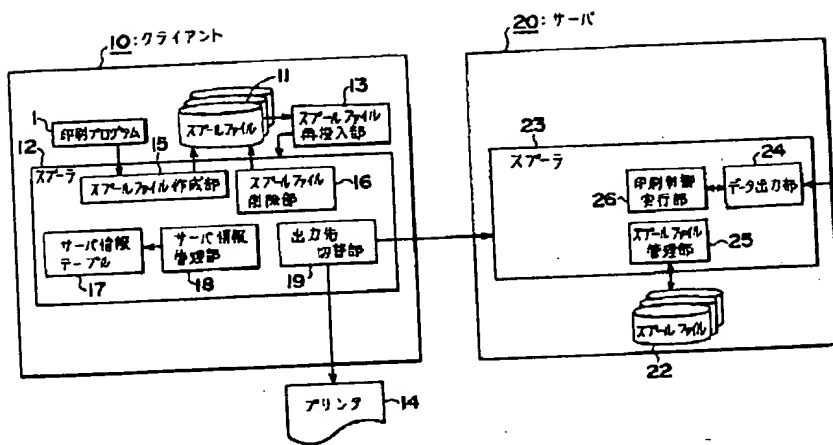
サーバ名	転送可能フラグ
サーバ1	1
サーバ2	0
サーバ3	1

【図3】

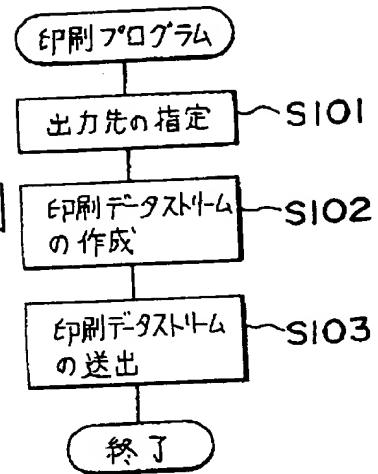
本実施の形態における正常時のシステムタイムチャート



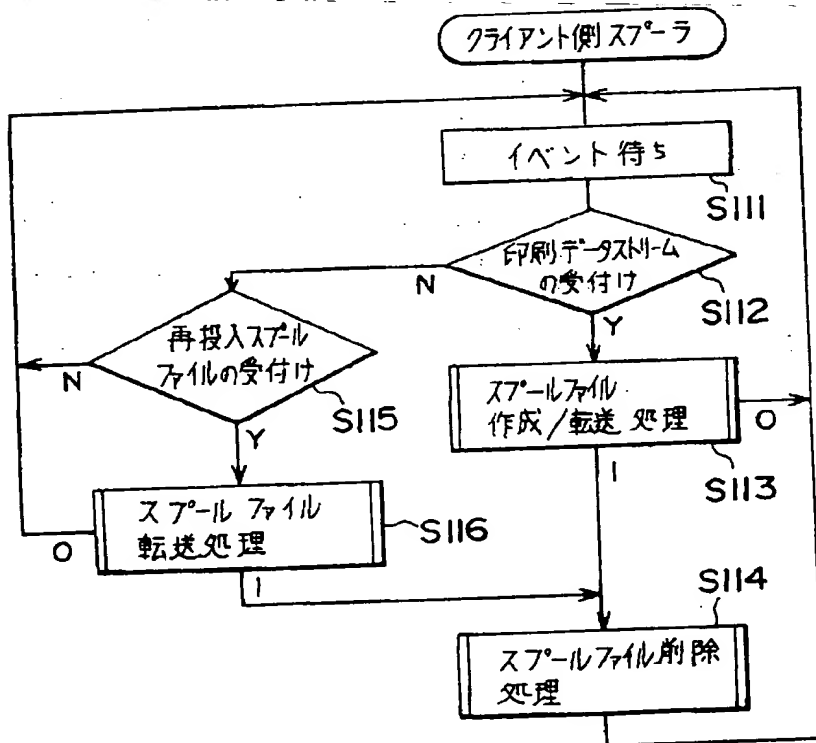
【図1】



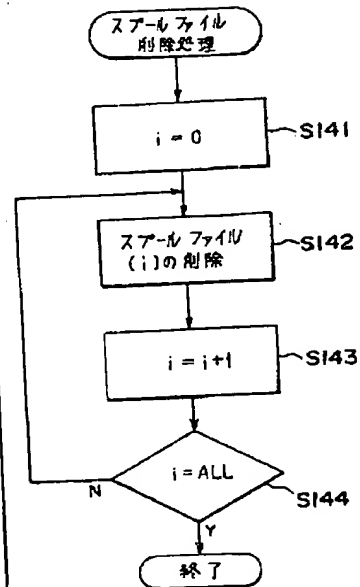
【図4】



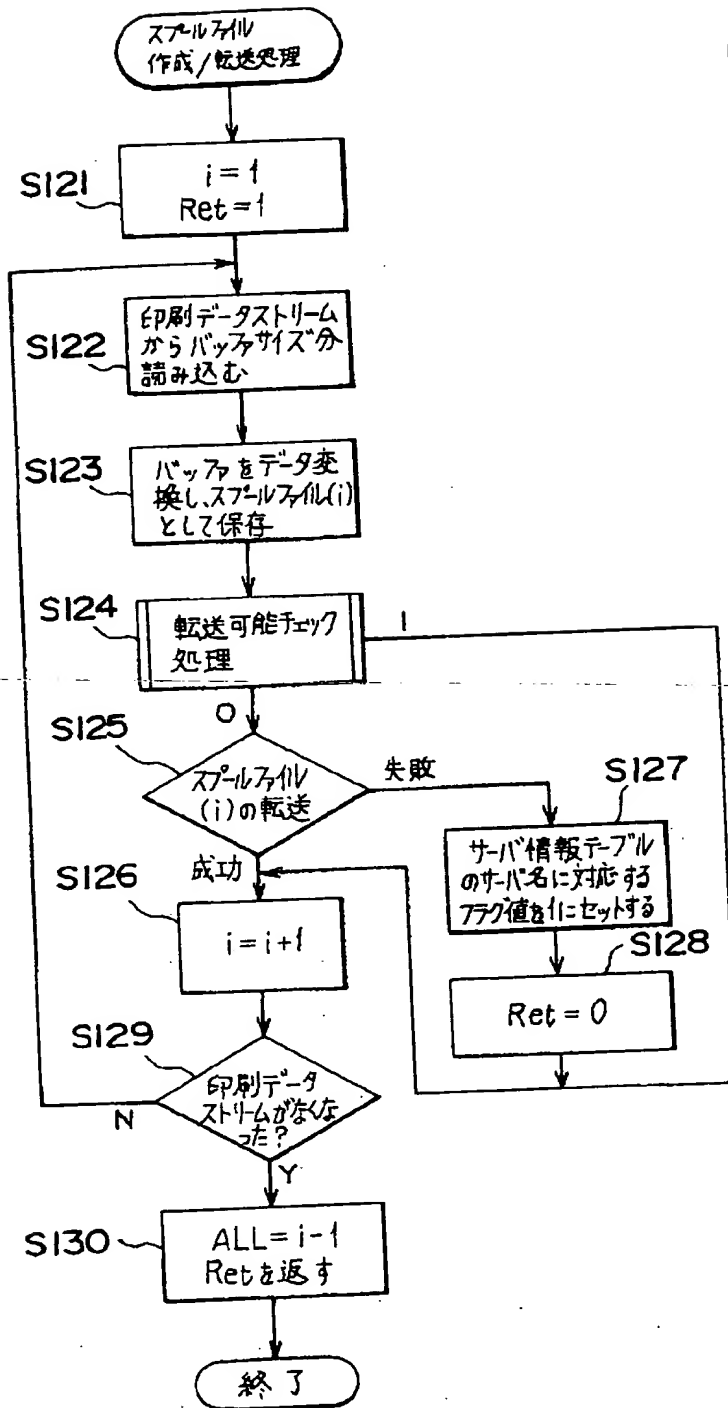
【図5】



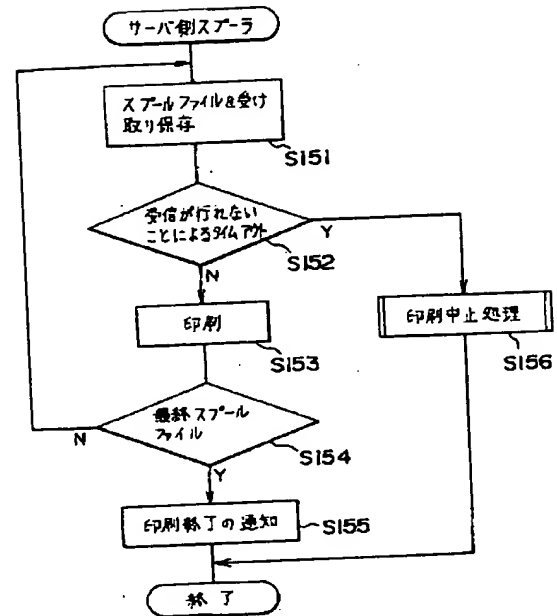
【図8】



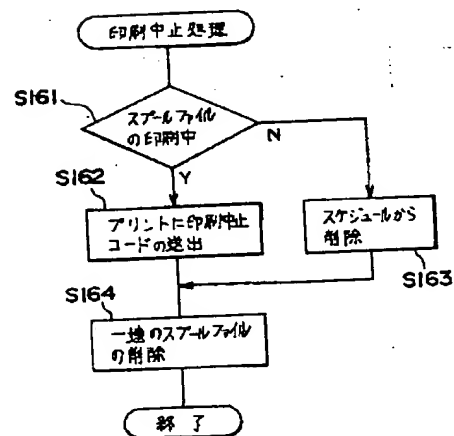
【図6】



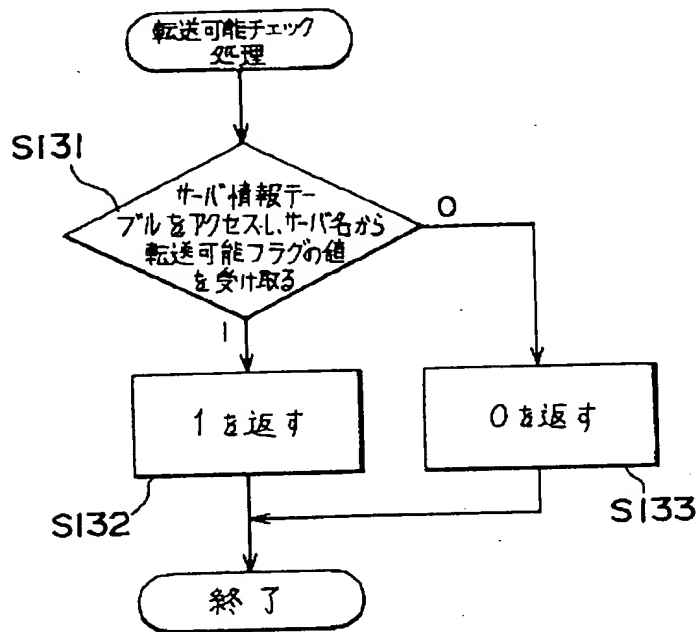
【図9】



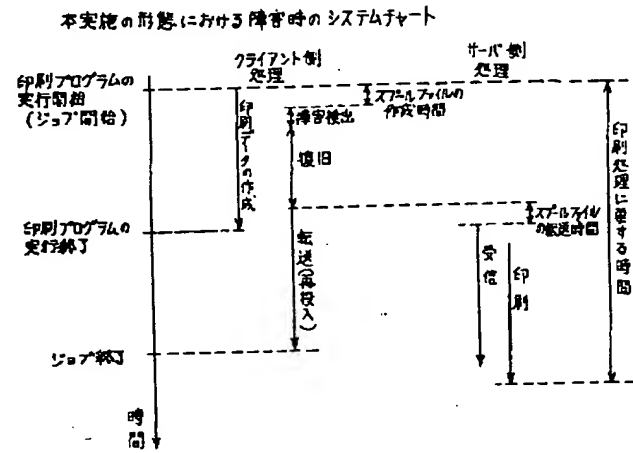
【図10】



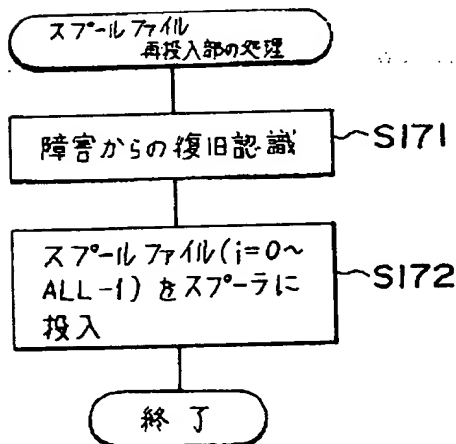
【図7】



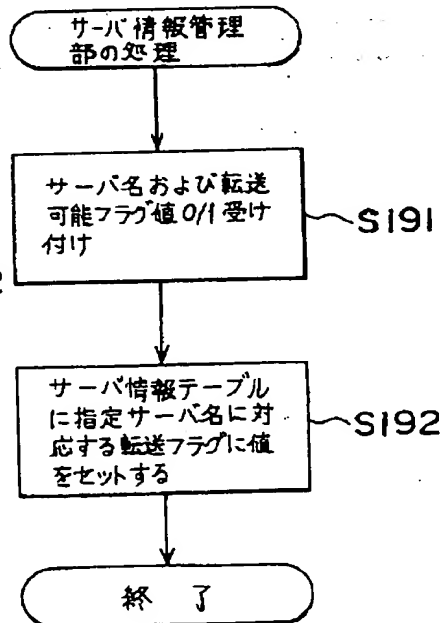
【図11】



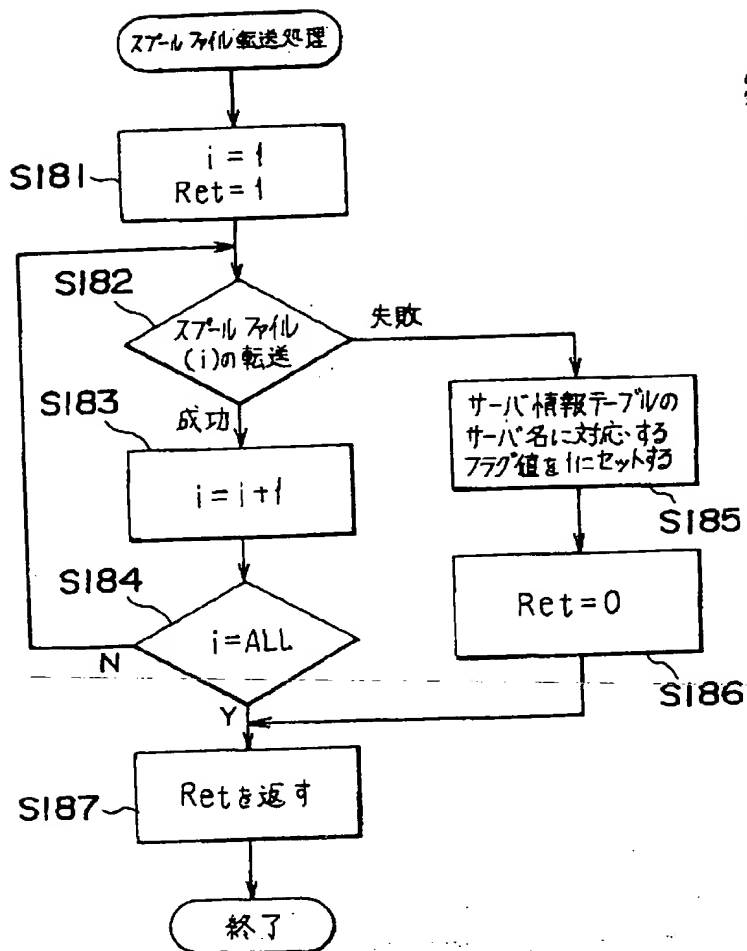
【図12】



【図14】

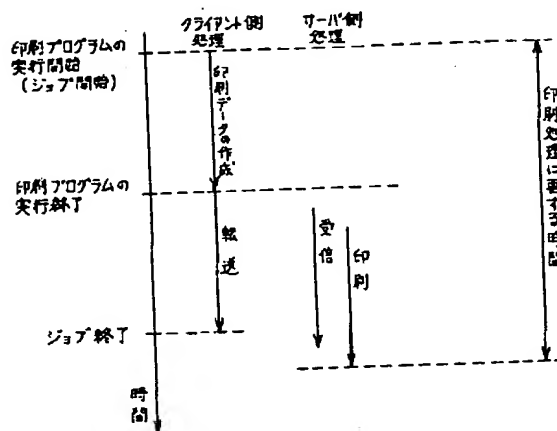


【図13】

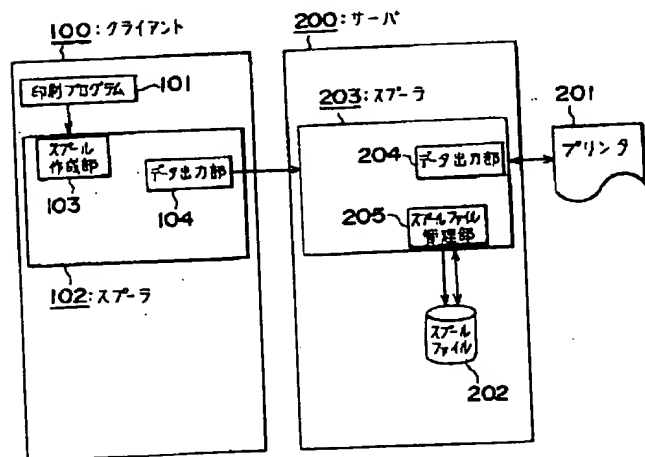


【図17】

従来における正常時のシステムタイムチャート

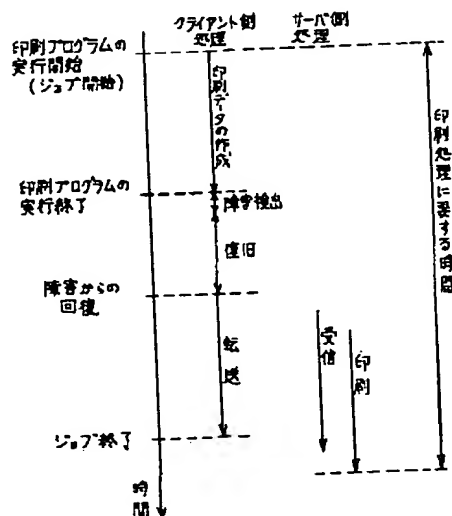


【図16】



【図18】

従来におけるネットワーク障害時のシステムタイムチャート



【図15】

